УДК 004.934

В.Г. Прохоров

АНАЛИЗ ПАРАМЕТРОВ КАК СРЕДСТВО ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ ОБУЧЕНИЯ НЕЙРОННЫХ СЕТЕЙ

Рассмотрены общие особенности обучения нейросетей методом обратного распространения ошибки. Проанализированы различные параметры сети, исследовано их влияние на эффективность обучения. Описаны изменения в ходе обучения нейронных сетей при модификации параметров.

Введение

Алгоритм обратного распространения ошибки является наиболее широкоиспользуемым при обучении нейронных сетей. Причиной этого является его концептуальная простота, относительно высокая вычислительная эффективность и способность к достижению результата в большинстве случаев. Тем не менее, настройка такого алгоритма для эффективной работы – трудоемкий процесс, который зачастую сложно описать языком формул. Проектирование и обучение сети с помощью обратного распространения ошибки требует выбора многих параметров: числа и типа узлов, слоев, скоростей обучения, обучающих и проверочных выборок и т.д. Эти параметры коренным образом влияют на дальнейшую работу сети, при этом, нет единого набора параметров, который бы гарантировал оптимальную работу, так как параметры зависят от специфики поставленной задачи.

Данная работа – результат исследования различных параметров сети, их влияния на скорость обучения и конечную эффективность работы. Здесь не рассматриваются алгоритм обратного распространения, ошибки и методы повышения сходимости второго порядка (методы Левенберга–Марквадта, использование численного метода Ньютона–Гаусса для оптимизации вычисления Гессиана) они описаны в [1, 2]. Основное внимание будет уделено описанию параметров, позволяющие разработчикам нейронных сетей принять правильные решения при проектировании и обучении.

Наиболее распространенный алгоритм обучения – обратное распространение ошибки, использует метод градиентного спуска для нахождения минимума и может работать крайне медленно в случае большого числа слоев персептрона, плоской поверхности решений, множеством локальных минимумов и других факторов. Не существует математической формулы, которая гарантирует сходимость сети в точке наилучшего решения, быструю сходимость, и даже то, что такая сходимость действительно произойдет. Далее будет рассмотрен ряд механизмов нейронной сети, настройка которых увеличивает шансы найти наилучшее решение, сократить время обучения, иногда на порядок.

1. Последовательное и пакетное обучение

Как известно, обучение нейронной сети происходит за счет корректировки весов. Эта корректировка может происходить на каждом шаге (последовательное обучение), или после прохода по всей обучающей выборке. В этом случае, изменения весов накапливаются и суммируются один раз. Такой метод обучения называют пакетным.

В большинстве случаев применяется последовательное обучение, так как оно значительно быстрее пакетного и чаще всего находит лучшее решение, чем пакет -

ное. Еще одно преимущество последовательного обучения – это возможность отслеживать изменения в сети при обучении на уровне связей. Именно поэтому на практике чаще всего используется метод последовательного обучения.

Тем не менее, в отдельных случаях пакетный метод может быть эффективно использован – ряд алгоритмов повышения сходимости второго порядка применим только к пакетному обучению, что обусловливает использование пакетного метода в случаях, когда время обучения сети является критичным. Это достигается за счет вычисления не только самого градиента обучения, но и кривизны поверхности, в которой происходит поиск минимума. Зная значение кривизны поверхности и градиента можно приблизительно рассчитать нахождение искомого минимума [3].

2. Анализ обучающей выборки

Особенность обучения нейронной сети в том, что она наиболее эффективно учится на незнакомых примерах. В этом случае, система получает больше новой информации, которая, как правило, изменяет направление градиента. Оценка информативности каждого примера является нетривиальной задачей. Есть несколько простых и эффективных эвристик, позволяющих обучать систему на информативных примерах. Одна из них – последовательное обучение на разных (т.е. принадлежащих к разным классам) примерах, поскольку принадлежащие одному классу примеры содержат похожую информацию.

Еще один способ оценки информативности примера – анализ его выходного вектора ошибок. Очевидно, что если при обучении сети на определенном примере, выходная ошибка имеет большое значение, то такой пример содержит много новой информации, не был заучен сетью и имеет смысл подавать сети такой пример чаще остальных. Заметим, что величина ошибки, при которой имеет смысл повторять изучение примера, носит относительный характер, и определяется через отношение к ошибкам других обучающих примеров.

Использование повторной подачи примеров на обучение может привести к низкой эффективности обучения сети. Рассмотрим следующий случай: допустим, идет обучение сети на множестве примеров, часть которых является искаженными. Очевидно, что такие примеры будут давать большую ошибку, следственно будут поданы на повторное обучение. Как результат, веса нейросети будут скорректированы по направлению к искаженным примерам, что является нежелательным. С другой стороны, использование такого метода является необходимым, когда обучающие примеры существенно отличаются количественно и система не может запомнить редко встречаемые примеры [4].

3. Нормирование значений

Сходимость сети значительно повышается при условии, когда среднее значение входных сигналов приблизительно равно нулю. Для иллюстрации данного утверждения рассмотрим случай, когда все значения входных сигналов положительны. Поскольку в процессе обучения веса, идущие к определенному нейрону, меняются на векторную величину, пропорциональную скалярной ошибке и входному вектору, то знак всех компонентов вектора, на который изменяются веса, будет знаком ошибки-скаляра. Как следствие, все веса могут одновременно или увеличиваться, или уменьшатся для определенного обучающего примера. Таким образом, если для нахождения минимума вектор градиент должен сменить направление, он будет менять его за счет сложной комбинации поворотов в противоположных направлениях, что значительно уменьшает скорость сходимости. Это явление имеет место, если все значения входных сигналов отрицательны. Именно по этой причине важно нормировать значения входных сигналов так, чтоб их среднее значение было близко к нулю.

Такой подход следует применять ко всем слоям нейронной сети, поскольку выходные значения одного слоя – входные значения для следующего слоя. В этом случае, такое нормирование проводит сигмоидальная активирующая функция, при-

меняемая в данной сети. Более подробно различные виды активирующих функций будут рассмотрены в следующем разделе данной работы.

Кроме нормирования значений компонента, так чтоб среднее значение было близко к нулю, необходимо также осуществить общее нормирование всех компонент, так, чтоб они принадлежали одному диапазону значений. Математически, это требование звучит следующим образом: ковариация всех входных сигналов должна быть примерно одинаковой, при этом ковариация вычисляется по формуле

$$C_i = \frac{1}{P}\sum_{p=1}^{P}(z_i^p)^2 , \qquad (1)$$

где P – общее число обучающих примеров, C_i – значение ковариации всех значений i-го нейрона входного слоя, а z_i^p – i-й компонент обучающего примера p. Такое нормирование значительно ускоряет обучение за счет балансирования темпов обучения весов, присоединенных к входному слою. Заметим, что данное нормирование не стоит проводить, если заранее известно, что определенные входные сигналы менее важны, чем другие. В этом случае нужно, наоборот, уменьшить диапазон значений малозначимых нейронов входного слоя, так, чтоб они меньше влияли на процесс обучения.

4. Выбор сигмоидальной функции

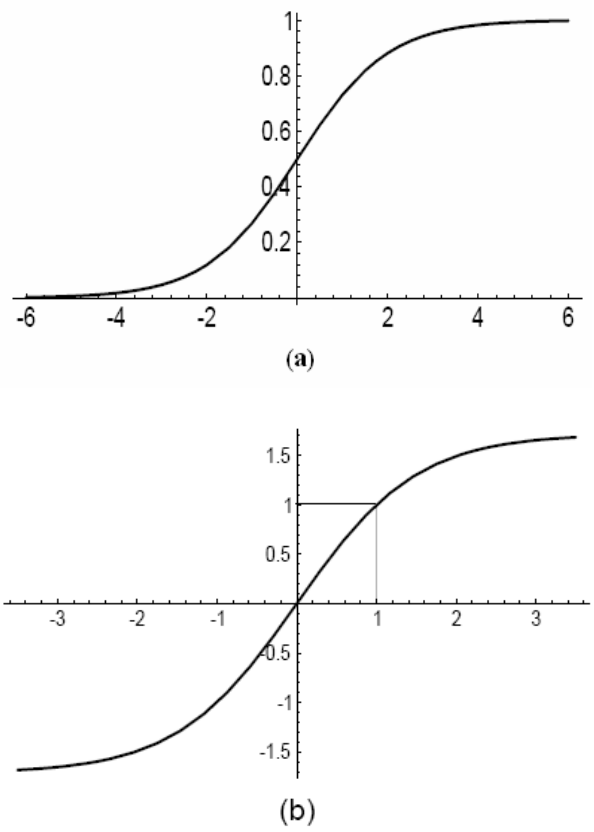
Нелинейные активирующие функции являются ключевым элементом нейросетевого механизма в силу их дифференцируемости, и свойству производить нелинейное преобразование входных данных. Чаще всего в этом случае применяются сигмоидные функции (сигмоиды) различных типов: монотонно возрастающие функции, которые при конечном значении аргумента стремятся к бесконечности. На практике применяются два вида таких функций – стандартная логистическая функция:

$$f(x) = \frac{1}{1+e^x} \qquad (2)$$

и гиперболический тангенс:

$$f(x) = \tanh(x)\,. \qquad (3)$$

На рисунке изображены графики этих функций: a) – логистическая функция, b) – гиперболический тангенс. Руководствуясь логикой нормализации значений, изложенной в предыдущем разделе, именно гиперболический тангенс является приемлемой активирующей функцией, так как среднее значение такой функции близко к нулю, что следует из симметричности функции относительно начала координат (в случае с логистической функцией, значения всегда позитивны). Напомним, что выходные значения активирующей функции являются входными значениями для следующего слоя, что повышает требования к функции, и диапазону возвращаемых ею значений.



Рисунок

На практике, применяется гиперболический тангенс с несколько видоизмененными параметрами (как показано на рисунке, диапазон значений несколько шире, чем [-1;1], стандартный диапазон гиперболическго тангенса). Наиболее при-

емлемым вариантом гиперболического тангенса является вариация

$$f(x) = 1.7159 \tanh(\frac{2}{3}x).$$

Такая функция обладает двумя важными особенностями $f(1)=1$, а вторая производная достигает своего максимума при $x=1$. Более подробно видоизмененный гиперболический тангенс рассмотрен в [5, 6].

5. Выбор целевых значений выходного слоя

В нейронных сетях, используемых для решения задач классификации, целевое значение функции, как правило, бинарное, например, [-1;1]. На первый взгляд, выбор в качестве целевых значений функции значения ее асимптот является наиболее логичным. На практике, у такого подхода есть ряд недостатков. Один из них – медленная сходимость нейросети. При обучении, нейросеть будет адаптировать веса так, чтоб они совпадали с целевыми значениям функции, а этого можно достичь только за счет асимптотичного приближения [7]. В результате, веса будут стремиться к большим значениям, но при этом производная функции при таких значениях будет стремится к нулю, что сделает изменение весов крайне малыми, т. е. изменение весов и процесс обучения фактически прекратятся. Заметим, что такой подход отчасти противоречит одному из основных требований к активирующей функции – нелинейности.

Еще один недостаток – низкая эффективность работы такой сети при распознавании неоднозначных объектов. Рассмотрим пример: на вход нейросети поступает объект, который лежит возле разделяющей 2 класса гиперплоскости. В идеальном варианте, сеть должна вернуть значение, лежащее между двумя целевыми значениями, т. е. далеко от обеих асимптот. Проблема в том, что большие веса нейросети, сдвигают результирующие значения к асимптотам сигмоида. Как результат, сеть может неправильно распознать определенный объект и при этом не сообщить никаких данных о том, что вероятность неправильного распознавания в данном случае велика.

Решение этой проблемы состоит в выборе таких целевых значений, которые лежат в пределах значений сигмоида, а не его асимптот. В этом случае, необходимо выбрать новые целевые значения так, чтоб значение активирующей функции не было ограничено линейной частью сигмоида. Выбор в качестве целевых значений точек максимума второй производной активирующей функции является наилучшей практикой – при таком подходе сохраняется требование к нелинейности. По этой причине, сигмоид б) на рисунке является наилучшим выбором активирующей функции. Его вторая производная имеет наибольшее значение в точках -1 и 1, что соответствует бинарным целевым значениям в задачах классификации.

6. Использование радиально базисных функций

Несмотря на то, что большинство систем формируют значение нейронов за счет скалярного произведения весов и сигналов, а также применения к результату такой операции сигмоидальной функции, можно использовать другие типы слоев и функций. Наиболее распространенным альтернативным вариантом является сеть на базе радиально базисных функций (РБФ). В сетях на основе РБФ скалярное произведение заменено эвклидовым расстоянием между входным сигналом и весами, а сигмоид заменен экспонентой. Значение функции для каждого выхода вычисляется по следующей формуле:

$$f(x) = \sum_{i=1}^{N} w_i \exp(-\frac{1}{2\sigma_i^2} \| x - v_i \|^2), \qquad (4)$$

где $v_i(\sigma_i)$ – стандартное отклонение i-го Гауссиана. РБФ могут как заменять стандартные нейроны, так и сосуществовать вместе в рамках разных слоев. На практике чаще всего применяется второй подход, например, современные сверточные сети с большой точностью распознавания используют РБФ при формировании последнего слоя нейронов.

В отличии от сигмоидов, которые определены всюду, отдельный РБФ нейрон

покрывает только небольшую локальную область входного пространства. Это определяет одно из преимуществ РБФ – во многих случаях небольшая область ускоряет адаптацию весов, т. е. обучение [8]. Использование РБФ в качестве базисных функций для моделирования входного пространства (вместо сигмоидов) также возможно, но целесообразность такой замены тесно связана с самой задачей. С другой стороны, небольшой размер РБФ области затрудняет ее использование в пространствах с большим числом измерений, так как для покрытия всего пространства необходимо значительное число нейронов. Поэтому, РБФ используют в последних уровнях нейросети (с малым количеством измерений), а сигмоиды – в уровнях с большим числом измерений.

7. Экспериментальная оценка эффективности различных параметров нейронных сетей

Прежде чем перейти к непосредственному описанию экспериментов и полученных результатов, стоит отметить, что эффективность обучения и работы нейронной сети зависит от предметной области, обучающей, и, в меньшей мере, проверочной выборки данных. Полученные в результате экспериментов результаты не следует воспринимать как абсолютную истину – эффективность тех, или иных подходов может существенно меняться при разных исходных условиях, что будет продемонстрировано далее при оценке эффективности РБФ функций.

Все эксперименты проводились на полносвязной нейронной сети с одним скрытым слоем. Сеть обучалась распознаванию рукописных цифр (т. е. количество выходных классов равно 10), которые поступали на вход в виде битового изображения 28x28 пикселов (таким образом, входной слой сети состоит из 784 входных нейронов). Число нейронов скрытого слоя выбрано равным 50. Кроме эксперимента, оценивающего эффективность активирующих функций, в качестве такого использовался гиперболический тангенс. Обучение проводилось последовательным методом (кроме сравнительного эксперимента с пакетным) на стандартном наборе рукописных цифр MNIST, используемом учеными и энтузиастами для оценки эффективности OCR систем во всем мире. Размер обучающей выборки – 60.000 символов, проверочной – 10.000. Графические образы цифр не проходили предварительную обработку (центрирование, фильтрацию, масштабирование). Во втором эксперименте с оценкой эффективности РБФ функций в качестве обучающей выборки брались сгенерированные средствами .NET образы букв латинского алфавита, нарисованные разными шрифтами. В этом случае, сеть состояла из 784-100-26 нейронов, обучающая выборка состояла из 3250 символов (125 полных алфавитов), проверочная – из 178 символов (3 алфавита из обучающей выборки с внесенными минимальными искажениями).

В качестве критериев эффективности работы брались такие параметры: скорость обучения, точность распознавания символов из проверочной выборки.

Всего проведено 5 сравнительных экспериментов, в каждом из них менялся параметр:

- последовательное и пакетное обучение;
- нормированные и не нормированные значения;
- гиперболический тангенс и логистическая функция;
- РБФ и стандартные активирующие функции;
- РБФ и стандартные активирующие функции (с другой обучающей выборкой – набором букв).

Результаты экспериментов приведены в таблице. Рассмотрим их подробнее.

Как отмечено выше, пакетное обучение стоит использовать только в тех случаях, когда для ускорения обучения используются алгоритмы второго порядка. В противном случае, последовательное обучение является более эффективным.

Логистическую функцию вообще не следует использовать для обучения нейросетей. Эксперименты подтверждают ее неэффективность по сравнению с гиперболическим тангенсом.

Нормирование входов – эффективный прием, который стоит применять для задач с однородными и одинаково важными для сети входными данными.

Использование РБФ для распознавания символов не превзошло классический подход. Однако, с увеличением числа выходных классов (с 10 до 26), преимущество имеют РБФ. Причины такого эффекта подробно описаны в [9].

Выводы

Сравнение различных параметров нейронных сетей, а также анализ их влияния на работу нейронной сети осуществляется на основе классической полносвязной сети, настроенной на распознавание образов. Многие параметры тесно связаны с самой задачей и ее предметной областью. По этим причинам цель работы – не поиск оптимального набора параметров, обеспечивающего наиболее быстрое обучение и точное распознавание, а подробный анализ параметров и их влияния на работу сети.

Отметим, что в данной работе пропущен анализ начального распределения весовых коэффициентов, а также вариации скорости обучения, которая описана в [7]. Начальное распределение весовых коэффициентов – открытый вопрос, что может стать темой отдельной статьи. Различные подходы к начальному распределению показаны в [5, 7].

Таблица. Результаты экспериментов

Первый параметр	Время обучения, час: мин: сек	Точность распознавания	Второй параметр	Время обучения, час: мин: сек	Точность распознавания
Последовательное обучение	5:10:19	86.48%	Пакетное обучение	5:39:07	82.78%
Нормированные входные значения	5:10:19	86.48%	Ненормированные входные значения	6:12:48	79.58%
Гиперболический тангенс	5:10:19	86.48%	Логистическая функция	6:28:15	73.37%
Стандартная активирующая функция	5:10:19	86.48%	РБФ	6:24:43	81.29%
Стандартная активирующая (распознавание набора букв)	0:40:37	73.5%	РБФ (распознавание набора букв)	0:29:06	80.3%

1. *Le Cunn Y., Bottou L., Orr G.B.* Neural Networks: Tricks of the trade, Springer. – 1998. – P. 1 – 5.
2. *Каллан Р.* Основные концепции нейронных сетей. – М.: Вильямс, 2001. – С. 80 – 196.
3. *Burges C.J.C.* A Method for Training Neural Network to Recognize Character Strings, AT&T Bell Laboratories. – 1992. – P. 1 – 8.
4. *Simard P.Y.* Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis, Microsoft . – 1998. – P. 23 – 24.
5. *Le Cunn Y.* Efficient BackProp, Speech and Image Processing Services Research AT&T Lab. – 1998. – P. 5 – 16.
6. *Vaillant R.* Localization of Objects in Images, Speech and Image Processing Services Research AT&T Lab. – 1994. – P. 1 – 13.
7. *Хайкин С.* Нейронные сети. Полный курс Изд. второе (испр.). Прэнтис Холл. – 2006. – С. 239 – 298 ; 308 – 315.
8. *Le Cunn Y., Bottou L., Haffner P.* Gradient Based Learning Applied to Document Recognition, IEEE Press. – 1998 – P. 4 – 12.
9. *Прохоров В.* Использование сверхточных сетей для распознавания рукописных символов // Проблеми програмування. – 2008. – № 2-3. – С. 669 – 674.

Получено 13.10.09

Об авторе:

Прохоров Валерий Георгиевич,
аспирант Института программных систем НАН Украины.

Место работы автора:

Институт программных систем
НАН Украины.
03187, Киев -187,
Проспект Академика Глушкова, 40.
Телефон: 80509713876.
E-mail: makumazan84@yahoo.com